

Oefening Factory

HoGent

Oefening : GuiAbstractFactory

- ▶ Een andere operating system definieert een ander uiterlijk, zoals buttons and labels.
- ▶ Wil een toepassing overdraagbaar zijn over meer look-and-feel standaarden, dan moeten de operating systems niet in de code zijn vastgelegd voor een bepaalde look-and-feel.

a	ScreenFactory
	+createScreen(type : String) : Screen

a	Screen
	+Screen() +addButton() : void +addLabel() : void +viewComponents() : void

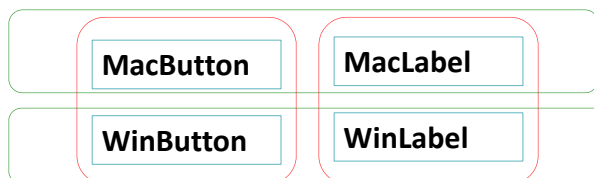
HoGent

Oefening : GuiAbstractFactory

- ▶ Afhankelijk van het operating system, passen we de factory aan, maar de “client implementation” blijft ongewijzigd.
- ▶ Indien er een nieuw operating system wordt toegevoegd dan dienen we een nieuwe factory en de button en de label te implementeren. We moeten de bestaande code niet wijzigen om een nieuw platform te kunnen toevoegen.
- ▶ Creëer een label en een button voor de operating systems Windows en Mac.

Oefening : GuiAbstractFactory

- ▶ Een familie van objecten = een verzameling objecten die samenwerken:
 - Familie Mac= MacButton + MacLabel
 - Familie Win= WinButton+ WinLabel
- ▶ Het gedrag van een object in één familie vind je ook terug in een object van de andere families:
 - gedrag MacButton == gedrag WinButton(show)
 - gedrag MacLabel == gedrag WinLabel (show)



Werkwijze

Werkwijze

1. Maak per **gedrag** een interface (Component)
2. Maak voor elke familie de concrete klassen aan
3. Maak een **abstracte factory** (GuiFactory). Voeg een create method toe per object soort
4. Maak per **familie** een concrete factory (WinFactory, MacFactory)
5. Injecteer de factory in de client